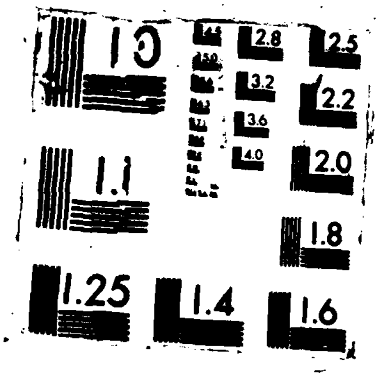
AD-A183 737 LANGUAGE MODELLING FOR AUTOMATIC SPEECH RECOGNITION: A 1/1
REVIEW(U) ROYAL SIGNALS AND RADAR ESTABLISHMENT MALVERN
(ENGLAND) L DODD ET AL. APR 87 RSRE-MEMO-4038
UNCLASSIFIED DRIC-BR-102355 F/G 5/7 NL


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6

UNLIMITED

2

RSRE
MEMORANDUM No. 4038

ROYAL SIGNALS & RADAR ESTABLISHMENT

LANGUAGE MODELLING FOR AUTOMATIC SPEECH RECOGNITION: A REVIEW

Authors: L Dodd and S R Johnson

PROCUREMENT EXECUTIVE,
MINISTRY OF DEFENCE,
RSRE MALVERN,
WORCS.

RSRE MEMORANDUM No. 4038

UNLIMITED

ROYAL SIGNALS AND RADAR ESTABLISHMENT

Memorandum 4038

TITLE : LANGUAGE MODELLING FOR AUTOMATIC SPEECH RECOGNITION : A REVIEW

AUTHORS : L Dodd and S R Johnson

DATE : April 1987

ABSTRACT

This memorandum reviews recent studies and developments in methods of language modelling which are specifically relevant to automatic speech recognition (ASR). An introduction is given to the general area of language models and the ways of formalising linguistic knowledge. Various techniques for applying phonological, syntactic and semantic constraints to ASR are discussed. The review covers papers written as early as the 1970's but the emphasis is on the more recent developments and techniques which are now being used in speech research. The formal methods of applying linguistic constraints are discussed and criticised according to their suitability for the speech research work carried out at RSRE.

CONTENTS

1. INTRODUCTION

In recent years automatic speech recognition (ASR) work at RSRE has concentrated mainly on recognition of isolated and connected words by comparing the patterns resulting from acoustic analysis of unknown utterances against statistical models of whole-word patterns, without reference to phonological, syntactic or semantic constraints. In order to improve the performance of current ASR systems and to enable the systems to work with large lexicons, it is desirable to include linguistic information in the form of phonological, syntactic and semantic constraints. These constraints are collectively referred to as linguistic constraints. The term "constraint" refers, in general, to information about what is (or is not) allowed, but it will be used also to mean what may be more (or less) likely to occur in the use of language.

A language model is a complete set of linguistic constraints expressed formally. In other words, language models are computationally useful formalisations of linguistic knowledge. At the syntactic level, a language model may be equivalent to a grammar. There are several alternative views on the nature of language and, in particular, the grammar of natural language. The early work in linguistics was concerned mainly with the development of syntactic grammar rules and phonological rules ("hard" constraints). More recently, there have been developments in stochastic grammars using probabilistic ("softer") rules to describe phonological and syntactic information. Other language models are based more simply on the probability of juxtaposition of words or phonemes (n-grams) where the probabilities are derived from the frequency of occurrence in very large samples of text.

One way of combining linguistic information with acoustic analysis is to use conditional probabilities and Bayes' rule. The work in ASR at RSRE uses the probability, $p(A|W)$, that when a speaker says the word (or sub-word unit or word string) W the acoustic evidence A will be observed. In order to estimate the probability, $p(W|A)$, that the utterance is W, given the acoustic evidence A, it is also necessary to calculate the probability that W would occur. These probabilities must be combined using Bayes' formula :

$$p(W|A) = \frac{p(W)\, p(A|W)}{p(A)}$$

A language model provides a method of computing a suitable probability, $p(W)$, for any proposed utterance [1]. In these terms, a language model is a mathematical formalisation of linguistic constraints which is used to predict the likelihood that any element from an allowed vocabulary will follow the string of elements in the utterance. In ASR, a language model may be used to limit the extent of the search for the correct word in an utterance, or to resolve ambiguity in the acoustic analysis.

Phonological information may be used to specify allowable sequences of phonetic segments (phonemes or syllables), predict the systematic variation in the pronunciations of words, or it may describe the role of prosody. For example, a phonological constraint will predict that /spl/ is an allowable sequence of phonemes in English, whereas /sbl/ is not. Lexical knowledge may be used to describe the internal structure of words in the language, or model the lexicon as a whole. A lexical constraint would, for instance, specify that although /blisk/ is phonologically allowable, it doesn't constitute an English word. Some examples of the use of phonological and lexical constraints are described in Section 2.

At a higher level, syntactic and semantic constraints can be applied. Syntactic constraints specify grammatical rules such as 'a sentence comprises a noun phrase followed by a verb phrase'. Syntactic constraints are used to reduce ambiguity in the analysis of a string of acoustic-phonetic data. For example, the utterance "Those two books are mine" would be analyised syntactically to eliminate the acoustic-lexical ambiguity in the second word in the sentence, which could be otherwise interpreted as either "two", "too" or "to". Section 3 covers the use of syntactic constraints in ASR.

Semantic constraints arise from the meanings of words, and combinations of words, and they help to resolve grammatical ambiguity. For example, semantic analysis of the sentence "John saw the man in the park with the fountain" would eliminate the syntactic ambiguity that the prepositional phrase "with the fountain" could be attached to either the noun phrase "the man" or "the park". Semantic constraints are most difficult to formalise. Some attempts are outlined in Section 4.

Examples of automatic speech recognition systems which employ some linguistic constraints are the HARPY system from CMU (derived from the HEARSAY and DRAGON systems), the IBM system, and BBN's HWIM. These are discussed briefly in Section 5.

In addition to the reference section, a bibliography is available, which is a compilation of useful articles, papers and books on the general subject of linguistic constraints in ASR.

2. PHONOLOGICAL CONSTRAINT MODELLING

Phonological constraints in ASR may take several forms. At a low level, the language-specific rules governing the permissible ways of combining phonemes into syllables and words may be used. At a different level, models of the alternative pronunciations of words, either those that are obligatory, conditioned by context (allophonic variation), or optional, conditioned by speaking rate, style, dialect etc. (phonological variation) may be constructed. In other cases information about the phonological structure of the dictionary (word length or frequency, syllabic structure, for example) may be used to aid lexical access, or to predict the performance of ASR. Alternatively, the suprasegmental features (prosodics) of the language may be modelled.

2.1. Phonological constraints

Most of the work in this area appears to be concerned with the specification of rules for alternative pronunciations of words (phonological transformation rules).

Barnett [2] describes a system developed at System Development Corporation, for describing and operating a set of phonological transformation rules. The goal of this rule system was to predict alternative pronunciations for lexicon entries. The models were specific to one speaker, or a small community of speakers sharing similar speaking characteristics. The rules were to be integrated and used in a prototype continuous speech understanding system (SDC SUS). Three sets of phonological rules were proposed - the first would generate the set of legal pronunciations realised by changes in phonemic spelling, the second was to model intra-syllabic co-articulation effects which may depend on phonetic features, and the third would model interactions over syllable and word boundaries. Only the first of these had been implemented at the time of the paper mentioned. Lexicon entries were phonemically spelled (using ARPABET) and have syllable and interior word boundaries marked, as well as stress levels.

As an example of these rules (which are fairly typical of most phonological transformation rules), in some varieties of English an unvoiced plosive may be inserted between a nasal and a following unvoiced fricative or plosive with a different place of articulation from the nasal, and the inserted plosive will have the same place of articulation as the nasal (homorganic stop insertion). For example, a [p] may be inserted between [m] and [th] in "something". The rule for this is as follows:

```
HOMORGANIC STOP INSERTION
(PLOS PLACE%1 -VOICE) = NASAL OPT*//(PLOS -VOICE) OR
                                    (FRIC -VOICE)
IF CLASS%3 EQ FRIC OR
PLACE%1 NQ PLACE%3
```

The first part is the rule name, the second defines the reconstruction possible, the third specifies left context (i.e. nasal plus optional syllable boundary), the fourth specifies right context (i.e. unvoiced plosive or fricative), and the fifth details any conditions (i.e. place of following plosive must be different from the place of the nasal).

The rules are unordered and deterministic as the aim is to generate the full set of alternative pronunciations for each entry.

When in use in the speech understanding system the lexicon contained 150 entries with an average length of 10 phonemes (including syllable and word boundaries). A set of 37 such rules generated an average of 2.3 alternatives per item.

More recently, Huckvale [3] has developed a set of rules as part of his Speech-Production Production System. His version of the insertion rule quoted above is given for comparison.

```
!Plosive Epenthesis
     [NASAL = NASAL] [*] [MANNER = FRICATIVE]
      -> [_] [MANNER :=STOP,PLACE := %1,VOICE:=%3] [_]
```

In the sequence 1. nasal, 2. null, 3. fricative insert a stop at 2. with the place of 1. and the voicing of 3.

A similar approach to applying phonological rules was taken by IBM Yorktown Heights [4]. Although their basic aims were the same, Cohen and Mercer were in addition concerned with associating a probability with each of the possible realisations of an utterance in an ASR system. Their main point was that some phonological variants of words/phrases are common to particular speaker populations and styles of speaking, and some are more frequently encountered than others. Therefore, it is necessary to associate with each utterance and pronunciation the probability that that pronunciation is produced as a realisation of that utterance. They appear to do this on a speaker-dependent basis, attaching speaker-dependent probabilities both to the base forms and to the phonological rules. An overall probability for that pronunciation is gained by combining these.

Their system consisted of a lexicon of phonemic base forms of American English, a set of context-sensitive phonological rules to account statistically for phonemic/allophonic variation resulting from idiolect/style/rate etc., and an algorithm for applying those rules to the base forms to generate variants. The base forms were represented as directed graphs, and the application of the rules produced an expanded graph which accounted for all possible pronunciations and their associated probabilities.

They did not mention how they discovered the probabilities for either the base forms or the rules. More recently, however, they have been obtaining better results with a much cruder set of networks, using Baum-Welsh to estimate the probabilities. They list all the rules for American English, along with some notes on their distribution, which although fascinating are unlikely to be particularly relevant to British English.

2.2. Phonotactic/lexical constraints

Phonotactic and lexical constraints are as important in speech recognition as higher level syntax or semantics. Lexical constraints can be used, for instance, to produce a "filter" which will exclude sequences which can not be words. Phonotactic constraints describe the allowable phoneme combinations of the language, and this is particularly useful when acoustic cues are missing or distorted. The importance of both this and lexical knowledge is clear from the fact that even trained phoneticians are poor at phonetically transcribing unknown languages for which they do not possess such knowledge.

An approach to modelling phonotactic constraints was described by Bourlard et al. in the context of recognition based on phonemic Markov Models [5]. The phonemes are characterised by very simple three-state Hidden Markov Models trained on connected speech. Recognition of words is done in one of two ways. In the first, a Markov model for each word is made by concatenating the phonemic models of its constituents, while in the second, the reference templates are the phoneme models and recognition is in two stages: recognition of a string of phonemes, then lexical look-up based on that string. The model of phonotactic constraints consisted of a simple "trained phonemic syntax". This was a boolean matrix, showing all possible phoneme pairs, so effectively forbidding illegal transitions.

Researchers at MIT are concerned with how knowledge of both phonotactic and lexical constraints might be used to help the recognition task [6]. They are examining large lexicons in order to investigate some of the properties of (American) English. Their investigations suggest two useful properties of the language which might usefully be exploited. The first has to do with the broad phonetic structure of words in the lexicon, and their work suggests that for isolated word recognition broad phonetic classifications of words would permit efficient lexical access, while being robust in the face of both allophonic and inter-speaker variations. However, it is not clear how to interpret this conclusion for the problem of continuous speech recognition, where word boundaries are usually not detectable.

The second property has to do with the use of prosodic information in lexical access, in that stressed syllables undergo less variation than do unstressed ones. Dictionary expansion by phonological rule, as described above, does not conventionally capture this aspect of phonetic variability, so it is necessary to find out the extent to which the stressed/unstressed distinction participates in lexical constraints. Studies of their large dictionary showed that more words can be uniquely identified by their stressed syllables than by the unstressed ones, so they conclude that recognition algorithms should perhaps not be too concerned with the identification of phones in unstressed syllables as these are more variable and less information bearing. This leads them to suggest that the stressed syllables should be represented in fine phonemic detail, while the unstressed ones can be described in broader terms.

However, recent work at Cambridge and Edinburgh has cast some doubt on the validity of both these conclusions. The Edinburgh team found that better results were obtained if the segments described in fine phonemic detail were selected randomly, rather than occurring only in stressed syllables [7]. They also point out that the MIT studies did not take proper account of word frequency information. For instance, although they quote the average equivalence class size as being just over 2, a lot of the most frequent words belong to classes considerably bigger than this.

A different approach to phonotactic modelling is being investigated at Bell-Northern Research, Canada [8]. Since the major allophonic variants of a phoneme are determined by its postition within the syllable, they have compiled a network of all possible English syllables. This is used in large vocabulary ASR in order to restrict the possible phoneme sequences to correspond to sequences of valid syllables. The syllable network was based on 60,000 phonemic transcriptions in Webster's 7th. Collegiate dictionary.

In order to model syllable-based allophonic variation they use a separate Markov source for each allophone depending upon its position in the syllable network. In experiments involving speaker-dependent isolated word recognition the unknown word is decoded as a sequence of syllables, where each syllable is a path through the network and each of the network's transitions is mapped onto a Markov source allophone model. Then statistical decoding is used to compute the most likely syllable sequences corresponding to words in their dictionary. As they are not using any higher level language model all the words are considered equally likely. They found that for isolated CVC words the use of separate Markov models for each allophone brought significant improvement over having only a single source for each phoneme. However, this improvement was not evident when the test set consisted of arbitrary words containing consonant clusters (perhaps due to undertraining?).

They also suggest that the phonotactic constraints in polysyllabic words might be further tightened by using a separate network for each syllable position within the word, as the number of valid syllables decreases with increasing syllable position in the word.

Kahn [9] expresses similar views as to the usefulness of syllabic structure to predict phonological variation in ASR, but suggests a rule-based approach. In addition, Kosaka and Wakita [10] show that there are syllabic structural differences between words with different frequencies of occurrence which might be exploited in lexical access for ASR.

Church [11] considers allophonic variation to be a rich source of contextual information, which should be exploited by ASR systems. Since allophonic variation is the result of predictable systematic linguistic processes it should provide important cues for the determination of word boundaries and stress assignment. For example, it is possible to use prosodic and rhythmic cues to indicate approximately where a word boundary will occur (for instance if there are two adjacent stressed syllables there must be a word boundary between them), but the precise location of the boundary can only be determined by using cues provided by the allophonic structure. He has implemented a chart parser (see Section 3.2) at the phonetic level to capture these allophonic rules. He also points out that a natural extension to this is a parsing mechanism based on simple matrix operations, where the entries in the matrix could be probabilities.

2.3. Other methods of using phonological knowledge in ASR

A number of approaches are based on performing some sort of preliminary analysis on the input in order to derive coarse, but reliable and easily extracted features, and then using higher-level information to guide a more detailed analysis where necessary.

Lea et al. [12], for example, propose the use of prosodic features to segment the continuous speech signal into phrases and sentences, and to locate stressed syllables (because these contain more reliable phonetic information). They think it important to do this in order to be able to make syntactic predictions at an early stage in the recognition process. After such a preliminary prosodic/acoustic analysis the lexical hypothesiser inserts words into the sentence structure, guided by contextual constraints (e.g. lexical categories which could occur at certain points in the sentence structure). Then acceptable syntactic/semantic constructs, based on information from the grammar and task domain, combine to form a total hypothesis. The sentence hypothesiser controls the order in which acoustic/phonetic patterns are generated for comparison with the input, and also determines when it is necessary to perform a more detailed phonetic analysis.

De Mori [13] also describes a rule-based system for the extraction of acoustic cues using a grammar of frames. The (speaker independent) rules take into account contextually conditioned constraints, bottom-up information and top-down prediction imposed by lexical constraints. The knowledge is shared among procedures acting as experts which co-operate to extract acoustic cues from the signal and to generate hypotheses about the bounds of syllabic segments, and the phonetic features inside those segments. A grammar of frames was chosen for this work because frames provide a means of integrating structural and procedural knowledge, and to handling context sensitive rules. They also provide a way of representing default knowledge, which can be used by the inference mechanisms. Procedures for extracting acoustic cues/features when necessary can also be easily implemented. Once a frame has been instantiated (by some event) the expert attempts to fill its slots, either by extracting features from the data, or by evaluating predicates, depending on the calculation of functions defined by the semantic attachments, or, if all else fails, by using default values. The alleged advantages of this approach are that it makes lexical access easier because it is being done on the basis of a few reliable and easily detected primary acoustic cues, and that the costly signal processing needed to extract the more detailed features need not be done on the whole of the utterance, but only on those parts where it is really necessary.

Becker and Poza [14] describe the acoustic processing in a syntactically guided Natural Language Speech Understanding System.

The purpose of the acoustic processor is to verify or reject hypotheses generated through the interaction of syntactic, semantic, and pragmatic information. The word verification sub-system contains a function for each word in the system, which takes as input the point in the utterance where the word is hypothesised to start and returns a value denoting the confidence that the word is actually there. These functions were written by a speech scientist using knowledge of acoustic phonetics. Spectrograms were used by the expert to aid in the determination of the characteristics of each word in a variety of contexts. This information is then used in order to decide which of the analysis routines to use in what order, and to determine the degree of confidence to be assigned.

Work in progress in CSELT, Turin [15], is also concerned with continuous speech understanding. Here the task is divided into two distinct stages. The first is concerned with recognition and uses no syntactic/semantic information. Using techniques based on Markov models, diphones are extracted from the continuous speech signal, and a lattice of scored word hypotheses is produced using only lexical and phonological knowledge.

This is then passed to the second, "understanding", stage, where the aim is to find the best scoring sequence covering the utterance. Because of the nature of the input to this stage, one of their needs is a parser which is tolerant of overlaps or small gaps between adjacent words, and of very short, unstressed words being missing. They are also concerned with developing a parsing strategy which is potentially parallel in operation. Therefore, they are using the chart parsing philosophy (see Section 3.2), as this combines the advantages of bottom-up and top-down processing, and also allows them to take into account the priorities of hypotheses derived from the word scores. Their implementation of the parser is based on an "Actor Network" where each actor contains syntactic rules (based on a dependency grammar), as well as semantic and lexical competence. Although syntactic and semantic information is kept strictly separate in order to retain flexibility, they stress the importance of using both types of knowledge in parallel, so hypotheses for sentence segments are created only if both sets of constraints are satisfied.

In addition to the probabilistic control of the parser which is based on the word scores from the lexical hypothesiser, they also see the need for heuristic control processes to control the search of incomplete hypotheses. The main effect of this heuristic control is to restrict the search space.

Another approach using Markov models is being investigated at Cambridge, where HMM techniques are being applied to define linguistic units at several levels [16]. The assumption is made that although the states of a HMM may not correspond to traditional linguistic units, they must be picking out something significant. Sub-word Markov models are obtained for the training words and the segments of speech corresponding to each state are extracted. These segments are then used to train a new model which will produce a transcription in terms of these new units.

At the grammatical level the intention is to deduce the grammatical units of the language from a very simple (children's learner reading books) vocabulary. The HMM is trained on phrases from that vocabulary and clustering techniques are used on the re-estimated transmission probability matrix in an attempt to determine relationships between elements of the vocabulary. They found quite a close correspondence between the resulting classes and traditional grammatical units.

3. SYNTACTIC CONSTRAINTS

The syntax of a string of symbols concerns the rules governing the arrangement of, and the interrelationships between, the elements in the string. A grammar is a set of rules which describe and govern the syntax of a sentence or of a string of data symbols. Grammar rules can be applied at any level (and to anything!) and can be used to generate strings or sentences which belong to the language described by the grammar rules. A parser is a mechanism for applying grammar rules in order to label elements in a string or to assess the validity of a string.

Syntactic analysis of text or speech falls into two parts which could be labelled structural and judgemental, respectively:

a. Deciding what the various segments of a sentence represent and how they relate to the rest of the sentence. That is, labelling (or tagging) words in a sentence with parts of speech.

b. Accepting (or not) a string of symbols (which could be data strings or English words, etc.) as valid according to the grammar rules. In the case of a stochastic grammar (see Section 3.3), the symbol string would be given a score which represents the likelihood that the string is a valid one.

There are many papers and books on the general subject of parsing (see Bibliography). The emphasis in this review paper, however, is on parsers which are being used for speech, rather than for text only. Techniques to apply syntactic constraints are usually in the form of a computer programming algorithm which parses a simplified (generally context-free) language. For example, Earley's Algorithm [17] is a particularly useful method for parsing context-free languages. Other more powerful techniques are now being developed which can handle the complexities of natural language (for example, Augmented Transition Networks, see Section 3.1).

Another approach used in speech recognition, which can be directly related to stochastic grammars and ATNs, is Chart Parsing [18] (see Section 3.2). There are very strong links between Stochastic Grammars and Chart Parsing which merit further investigation.

3.1. Augmented Transition Networks

An Augmented Transition Network (ATN) [19] is a very useful way of representing complex grammar structures in the form of a network (or connected sub-networks) [20]. BBN have successfully used ATN's in their speech recognition work [21]. An ATN allows recursive calls to other networks (or sub-networks) in the overall system. The facility which gives it the power to represent natural language is in the form of changeable registers which can be continually checked and updated and which can transfer control to various parts of the network. (The latter gives ATNs the power of a Turing machine.)

As an example of what is typically held in the registers, the word "John" found at the beginning of a sentence causes the SUBJ register to contain the proper noun phrase "John". If the sentence is then found to be passive "John" would be moved to the OBJ register. Other registers such as TNS and TYPE, hold information about the verb tense and the type of sentence (e.g. declarative or question).

3.2. Chart Parsing

Charts provide a data structure for parsing. They are designed to handle the inherent ambiguity of natural language efficiently. A chart is simply a directed graph with each arc (or edge) representing a node (vertex) in the analysis of a string. Initially, all the nodes are pre-terminals (i.e. represent words), and non-terminal edges are added when adjacent edges can be combined into larger structures according to the grammar. Each string bounded by an edge is called a 'well-formed substring'. If the string can be parsed there will be an edge which accounts for the whole of the string.

The notion of 'well-formed substring' is a key one in chart parsing, as it is this which allows the treatment of ambiguity. If, for instance, the goal was to analyse the sentence "John saw the man with the telescope" there is one possible parse where the prepositional phrase is attached to the object noun phrase, and another where it is attached to the verb phrase directly. In a chart parser, because intermediate structures such as noun phrase, and prepositional phrase are stored as well-formed substrings, they do not need to be rebuilt each time a different parse is found, as they would in an ATN.

Since chart parsers are data structures for parsing, they may be used to implement different types of grammar, and can be run in either top-down or bottom-up mode. Church [22] proposes a parser based on matrix operations, where the chart is decomposed into a set of binary matrices, one for each part of speech, indexed by a pair of positions in the string of symbols being parsed. An entry in the matrix is 1 if the chart has a constituent for that part of speech spanning that entry, and 0 if it has not. If a stochastic grammar were being parsed, the entries in the matrix would be probabilities, rather than 1s and 0s.

3.3. Stochastic Formal Grammars

In general, a Formal Grammar can be used, in a mathematical sense, to describe natural and unnatural languages, such as programming languages. Formal grammar rules (usually called rewrite or production rules) are used to generate sentences in a language or patterns of data. In a Stochastic Formal Grammar (SFG) every production rule (e.g. NP -> Det,Noun) has a probability associated with it such that the probabilities for all production rules with the same left hand side must sum to unity [23].

Stochastic formal grammars can be directly related to Hidden Markov Models [24], which are now widely used in ASR. Using Baker's "nodal span" principle it is possible to generalise the algorithms for tuning and using HMMs so that they can be applied to Stochastic (Formal) Grammars. Baker's Inside/Outside algorithm [25], is an automatic technique for re-estimation of production rule probabilities. The algorithm is an extension of the Forward/Backward algorithm [26] which is used to re-estimate probabilities in Hidden Markov models. IBM are using the Inside/Outside algorithm in their current research on language modelling in ASR [1].

4. SEMANTIC CONSTRAINTS

Semantic information may be important in resolving ambiguities that cannot be resolved at the acoustic, lexical, or syntactic levels. For example, the word "plane" in "I am hoping to catch the eleven o'clock plane" is ambiguous acoustically (it is homophonous with "plain"), lexically (it has two possible meanings -'means of transport' or 'tool') and syntactically (it is being used as a noun). Semantic information would specify that the more likely interpretation in this case would be 'means of transport'. The alternative interpretation is not impossible, just rather unlikely.

However, the amount of information that would need to be incorporated in a system to enable semantic analysis of unrestricted natural language is huge, and no clear boundary exists between such information and the pragmatic knowledge about the world used in general problem solving. For this reason most research on semantic analysis has been done in the field of Artificial Intelligence, and tends to be application-specific [27,28]. In addition, not much progress has been made in developing formal techniques for semantic analysis which can be directly applied to established methods in automatic speech recognition, so most of the early work used natural language text input to highly specified domains. The computer programs such as SHRDLU [29], STUDENT [30], SIR [31] and TLC [32] use rules which are specific to the vocabulary and subject matter, and they are generally complex and lengthy (usually written in LISP).

Work in the 1960s demonstrated the problems of using limited logic systems and the need for a more general, but formal, approach to storing and processing the complex information associated with semantics (for example Semantic Nets). Later developments in declarative AI languages, and propositional and first-order predicate logics seemed to offer a way of establishing a 'deductive' method of interpreting sentences. The problem is that such logics rely on compact sets of neatly defined logical axioms for their deductive procedures and so cannot deal with the wide set of heuristic (general problem solving) procedures which may be necessary in understanding natural language. However, recent work on Definite Clause Grammars [33] has its origins in such theories.

Early literature in the field showed a strong divide between the various areas of linguistics, with many researchers seeing a clear boundary between syntax and semantics, and the majority considering the former to be of prime importance (hence the advanced state of syntactic methods relative to semantic ones). Others were primarily concerned with semantic information processing, systems such as Schank's Conceptual Dependency [34] and Wilks' Preference Semantics [35] being typical examples of these. However, more recently the divide between syntax and semantics based analysis has been narrowed by work in which the distinctions between then are not seen as clear cut, and both sorts of information are considered equally important. The communicative aspect of language use is now emphasised, so rather than trying to discover what patterns there are in language,and then finding out what those patterns mean (the traditional 'syntactic' approach), the issue is one of discovering how language is patterned to convey meaning (the 'functional' approach). This different perspective has been a major influence in the development of Lexical Functional Grammars [36], Functional and Systemic Functional Grammars [37,38], Case Grammars [39] and Word Grammars [40] (to name but a few!)which take a more unified approach to language understanding. In addition, early work with semantic nets, such as that of Quillian [41] on Semantic Memory may now be applicable to recent developments in "connectionist" methods [42], so may provide a formal means of integrating semantic and syntactic constraints by exploiting parallel, rather than serial, processing. Such efforts may lead to the development of a more formal set of methods for applying semantics (in conjunction with syntax) to ASR.

5. ASR SYSTEMS THAT INCLUDE LANGUAGE MODELLING

5.1. HEARSAY II

HEARSAY II [43] was the first example of the "blackboard" approach to the organisation of large quantities of linguistic information. The system consisted of sets of independent modules (knowledge sources), each containing domain specific knowledge (phonetic, syntactic, semantic, pragmatic), and a shared data structure, called a blackboard, through which hypotheses from the knowledge sources could be accessed and modified as necessary. The acoustic-phonetic and phonological components were feature-based rewrite rules, while the syntactic component generated hypotheses based on the probabilities of occurrence of grammatical constructs. The semantic domain was restricted to retrieval of daily news stories. The main problem with this approach was that the scheduling of events was extremely complex, as at any particular point in the analysis the choice of which of a large number of potentially applicable knowledge sources to activate had to be made.

5.2. DRAGON-I

The Dragon system [44] was based on a finite-state network representation and the techniques used are Hidden Markov Models and Dynamic Programming (DP). Global optimality is guaranteed by the DP search because each possible path through the state network is covered. Combinatorial explosion is avoided by recombining alternative paths as frequently as new paths are added to the search space. In fact, the number of computations is linear in the length of the utterance.

The optimality depends on the finite-state grammar assumption and problems could occur when trying to model more complex (context sensitive) grammars. However, Baker claims [44] that the distinction between finite-state and higher order grammars is somewhat artificial and that the issue is one of modelling as accurately as possible the conditional probabilities (estimated by the frequency of acoustic events in the sequence, rather than by the frequency of words), not one of generating the proper language or grammar.

5.3. HARPY

The Harpy system [45] is an extension of the Dragon system which also incorporates features of the Hearsay II system. Unlike the Dragon system not all paths through the network are searched; only those paths which would be considered "near misses" according to the grammatical constraints are actually pursued. It uses the same kind of highly constrained finite-state grammar as Dragon-I.

5.4. HWIM

The Hear What I Mean (HWIM) system [46] was BBN's second major ASR system (the first being SPEECHLIS in the context of the natural language front-end called LUNAR). It was developed to handle travel budget manager tasks and so the vocabulary is limited by the particular application. The system uses a middle-out "island - driven" parsing strategy which incorporates Augmented Transition Networks (ATN's) (See Section 3.1). The modelling of phonetic, lexical, syntactic, semantic and pragmatic constraints uses a series of Cascaded ATN's [47].

A major concern in ASR systems is finding a suitable method of accessing words in the lexicon which are acoustically similar to labelled phonetic segments in order to arrive at the most likely interpretation of an utterance. A problem occurs when the number of predicted words is greater for one interpretation of the utterance than for another and hence there are different numbers of likelihood scores which need to be combined to give an overall score for each utterance. Therefore the scores of the different word hypotheses need to be normalised in some way.

5.5. THE IBM LANGUAGE MODEL

The language model of the current (1985) isolated word recogniser of the IBM (Yorktown Heights) research group [48,49] simply combines the probabilities of short sub-strings of words (typically tri-grams). To avoid extremely large numbers of possible combinations of words syntactic tags are used and the words are organised into various equivalence classes. The major simplification of the IBM system is that the speech input is in the form of isolated words and so the problem of finding the word boundaries is obviated.

The tri-gram modelling approach

$$P(w_3|w_2w_1)=q_3f(w_3|w_2w_1)+q_2f(w_3|w_2)$$
$$+q_1f(w_3)$$

where $q_1 + q_2 + q_3 = 1$ and $q_1 + q_2 + q_3 >= 0$

has the following problems :-

a. Non-occurrence of the trigram, bigram or unigram in the training text (i.e. the need for a very large training text). This problem is overcome by using a smoothing technique whenever the frequency counts are zero [49].

b. However large the training text, the resulting trigram model is always dependent on the context of that training text.

c. Inclusion of determiners and other common words in the trigram probability frequency counts can lead to loss of linguistic information. For example, the probability of the word "issues" following "resolve all" is very high but, if the phrase to be recognised is "resolve all the issues", then the word "issues" has a low probability as the third word of the trigram beginning "all the". (IBM are considering ways of addressing this problem.)

Part of speech (POS) classification is to be included as an additional term in the above equation in the next generation of IBM recognisers in the following form :

$$k(w_3|g_3)\ h(g_3\ |\ g(w_2)\ g(w_1))$$

where $g(w_i)$ is the POS class of the word w_i and g_3 is the POS class to be assigned to the word w_3 to be predicted. The probabilities k and h are estimated using the Forward-Backward (FB) algorithm on the training text [48].

The POS classifications are not based on traditional POS labels (e.g. noun,verb,etc). The POS classes are labelled by a representative word (called a "nucleus") which is a frequently used or grammatically important word (at present there are about 200 representative words). The FB algorithm is used to "organise" the remaining words in the lexicon into classes which are labelled by the representative words. As an example, all numbers form one class and all first names form another which is shared by titles Mr, Mrs, etc..

The main advantages of this approach are that some semantic information is implicit in the nuclear POS class and that the probabilities of a word belonging to any particular class can be re-estimated automatically. The major disadvantage is that the nuclear POS classes are entirely dependent on the context of the training text.

6. CONCLUSIONS

The need for the application of linguistic constraints at all levels is being recognised by those involved in the development of advanced automatic speech recognition systems. A number of current commercially available systems (e.g. DRAGON, IBM) are exploring the areas of applying syntactic constraints. The IBM system is also based on phonological rules.

Current methods at RSRE are concerned with Hidden Markov whole-word Models, and a natural extension of this work, to explore syntactic constraints, is to apply HMM's at a grammatical level. One method would be to assume that spoken utterances can be described by (probabilistic) context-free grammar rules. Then algorithms such as the I/O could be used to re-estimate the production rule probabilities as more examples of spoken utterances are input to the ASR system.

A problem with this approach might be the limitations imposed by the use of context-free grammars as models for spoken English. It is difficult to devise a thorough and rigorous model of spoken natural language by means of a context-free (or any other) grammar. However, it may not be necessary to use anything more sophisticated than a simple (regular) grammar for some of the applications of ASR systems. Nevertheless, speech to text systems (as may be used over telephone communications networks) will require a comprehensive model of natural language which will probably include semantic, pragmatic and prosodic information. Therefore, in the future it may be necessary to develop parallel processing methods to incorporate and use all or some forms of linguistic information simultaneously.

The majority of natural language analysers have been designed to handle text input. Speech, however has an important additional source of information, prosodics, which if properly exploited, could be of great benefit in automatic recognition systems.

REFERENCES

[1] F. Jelinek. "Markov Source Modeling of Text Generation." NATO Advanced Study Institute - Impact of Processing Techniques on Communications. Martinus Nijhoff 1985.

[2] J.A. Barnett. "A Phonological Rule Compiler." IEEE Symp. Speech Recognition, Carnegie-Mellon Univ. Pittsburgh, Pa. April 1974.

[3] M.A. Huckvale. "SP/PS: Speech-Production Production System." Six Monthly Report Three, MoD University Research Agreement 2047/0104/RSRE. "Phonetic Effects of Word Boundaries in Automatic Speech Recognition." Feb. 1985.

[4] P.S. Cohen & R.L. Mercer. "The Phonological Component of an Automatic Speech Recognition System." IEEE Symp. Speech Recognition, Carnegie-Mellon Univ. Pittsburgh, Pa. April 1974.

[5] R. Bourlard, Y. Kamp & C.J. Wellekens. "Speaker Dependent Connected Speech Recognition via Phonemic Markov Models." Proc. ICASSP-85; Florida, pp.1213-16. 1985.

[6] D.P. Huttenlocher & V.W. Zue. "Phonotactic and Lexical Constraints in Speech Recognition." Speech Communication Group Working Papers Vol.III, Fall 1983, Research Lab of Electronics. 1983.

[7] G.T.M. Altmann. "Lexical Stress, Lexical Discriminability and Partial Phonetic Information in Automatic Speech Recognition." Proc. Institute of Acoustics Conference on Speech and Hearing (Windermere). Vol. 8, Part 7, pp.471-8. 1986.

[8] V. Gupta, M. Lennig, J. Marcus & P. Mermelstein. "Syllable Network for Phonemic Decoding." Montreal Symposium on Speech Recognition, McGill University, Montreal. July 1986.

[9] D. Kahn. "Syllable-based Phonological Rules and their Implications for Speech Recognition." Montreal Symposium on Speech Recognition, McGill University, Montreal. July 1986.

[10] M. Kosaka & H. Wakita. "Syllable Structure of English Words: implications for Lexical Access." Montreal Symposium on Speech Recognition, McGill University, Montreal. July 1986.

[11] K.W. Church. "Allophonic and Phonotactic Constraints are Useful." Proc. IJCAI, pp.636-58. 1983.

[12] W.A. Lea, M.A. Medress & T.E. Skinner. "A Prosodically Guided Speech Understanding Strategy." IEEE Trans. Acoust. Speech and Signal Process. Vol. ASSP-23, pp.30-38. 1975.

[13] R. De Mori. "Extraction of Acoustic Cues using a Grammar of Frames." Speech Communication 2, pp.223-225. North-Holland. 1983.

[14] R.W. Becker & F. Poza. "Acoustic Phonetic Research in Speech Recognition." IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP-23, No.5, 1975.

[15] R. Gemello & C. Rullent. "Continuous Speech Understanding." CSELT Tech. Rept. (Italy) Vol.14, No.2, pp.117-20. 1986.

[16] R. Nag, S.C. Austin & F. Fallside. "Using Hidden Markov Models to Define Linguistic Units." Proc ICASSP-86; Tokyo. pp.2239-42. 1986.

[17] J. Earley. "An Efficient Context-free Parsing Algorithm." CACM, Vol.13, No.2, pp.94-102. Feb. 1970.

[18] H. Thompson & G. Ritchie. "Techniques for Parsing Natural Language: Two examples." D.A.I. Research paper 183, University of Edinburgh. (Also in O'Shea & Eisenstadt (Eds.) Artificial Intelligence Skills. Harper and Ross) 1983.

[19] W.A. Woods. "Transition Network Grammars for Natural Language Analysis." Computational Linguistics, CACM 13(10), pp.591-606. October 1970.

[20] J.E. Hopcroft & J.D. Ullman. "Formal Languages and their Relation to Automata." Reading, Mass: Addison-Wesley. 1969.

[21] D. Bobrow & J.B. Fraser. "An Augmented State Transition Network Analysis Procedure." Proceedings of International Joint Conference on Artificial Intelligence. Bedford Mass. Mitre Corp. pp. 557-568, 1969.

[22] K.W. Church. "A Finite-state Parser for Use in Speech Recognition." Proc. Assoc. Comp. Linguistics, pp.91-7. June 1983.

[23] King-Sun Fu & T.L. Booth. "Grammatical Inference: Introduction and Survey - Parts I and II." IEEE PAMI-8 No.3. May 1986.

[24] J.S. Bridle & L. Dodd. "Formal Grammars and Markov Models." RSRE Memo. To be published Jan. 1987.

[25] J.K. Baker. "Trainable Grammars for Speech Recognition." In D.H. Klatt & J.J. Wolf (Eds.) Speech Communication Papers for the 97th Meeting of the Acoustic Society of America, pp.547-550. 1979.

[26] J.K. Baker. "Stochastic Modeling for Automatic Speech Understanding." In D.R. Reddy (Ed.) Speech Recognition. Academic Press 1975.

[27] D. Michie (Ed.). "Machine Intelligence 3." New York: American Elsevier. 1968.

[28] C. Hewitt. "PLANNER: A Language for Theorem Proving in Robots." Proc. IJCAI 1969, Bedford, Mass:Mitre-Corp. pp.295-301. 1969.

[29] T. Winograd. "Understanding Natural Language." Academic Press. Also in Cognitive Psychology 3. No. 1. 1972.

[30] D. Bobrow. "Natural Language Input for a Computer Problem-solving System." In M. Minsky (Ed.), Semantic Information Processing. MIT Press 1968.

[31] B. Raphael. "SIR: Semantic Information Retrieval." In M. Minsky (Ed.) Semantic Information Processing. MIT Press. 1968.

[32] M.R. Quillian. "Teachable Language Comprehender: a Simulation Program and Theory of Language." Comp. Ling. CACM Vol.12, No.8, pp.459-76. Aug 1969.

[33] F.C.N. Pereira & D.H.D. Warren. "Definite Clause Grammars for Language Analysis - a Survey of the Formalism and a Comparison with Augmented Transition Networks." Artificial Intelligence Vol.13, No.3 pp.231-78. 1986.

[34] R.C. Schank. "Identification of Conceptualisations Underlying Natural Language." In Schank & Colby (Eds.) Computer Models of Thought and Language.San Fransisco:W.H. Freeman and Co. 1973.

[35] Y. Wilks. "An Intelligent Analyser and Understander of English." Communications of the ACM, Vol.18, No.5. 1975.

[36] R.M. Kaplan & J. Bresnan. "Lexical Functional Grammar: a formal system for grammatical representation." In Bresnan (Ed.), Mental Representations of Grammatical Relations. Cambridge MA:MIT Press, pp.173-281. 1982.

[37] S.C. Dik. "Functional Grammar." North-Holland Linguistic Series 37. 1978.

[38] M.A.K. Halliday. "Introduction to Functional Grammar." London: Edward Arnold, 1985.

[39] C. Fillmore. "The Case for Case." In E. Bach & R.T. Harms (Eds.) Universals in Linguistic Theory, New York:Holt. 1968.

[40] R.A. Hudson. "Word Grammar." Blackwell. 1984.

[41] M.R. Quillian. "Semantic Memory." In M. Minsky (Ed.) Semantic Information Processing. pp.281-309. MIT Press. 1968.

[42] D.L. Waltz & J.B. Pollack. "Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation." Cognitive Science 9, pp.51-74. 1985.

[43] V.R. Lesser, R.D. Fennell, L.D.Erman & D.R. Reddy. "Organization of the HEARSAY II Speech Understanding System." In Working Papers in Speech Recognition - III, CMU Computer Science Speech Group, April 1974.

[44] J.K. Baker. "The Dragon System - An Overview." IEEE Trans. on ASSP 23 No.1, February 1975.

[45] B.T. Lowerre. "The HARPY Speech Recognition System." In W.A. Lea (Ed.) Trends in Speech Recognition. Prentice-Hall, pp. 340-360, 1980. (Also Ph D Thesis Carnegie-Mellon University - Computer Science Department, April 1976.)

[46] J.J. Wolf and W.A. Woods. "The HWIM Speech Understanding System." In W.A. Lea (Ed.) Trends in Speech Recognition. Prentice-Hall, pp.316-339. 1980.

[47] W.A. Woods. "Language Processing for Speech Understanding."In F. Fallside & W.A. Woods (Eds.) Computer Speech Processing. Prentice-Hall, pp.305-334. 1985.

[48] F. Jelinek, R.L. Mercer & L.R. Bahl. "Continuous Speech Recognition: Statistical Methods." Handbook of Statistics, Vol.2, pp.549-573, 1982.

[49] F. Jelinek. "Self-Organised Language Modeling for Speech Recognition." IBM Yorktown Heights 1985.

END
9-87
DTIC